Image Partial Blur Detection and Classification

LIU, Renting

A Thesis Submitted in Partial Fulfillment
of the Requirements for the Degree of
Master of Philosophy
in
Computer Science and Engineering

June 2008

Thesis/Assessment Committee

Professor WONG Tien Tsin (Chair)

Professor JIA Jiaya Leo (Thesis Supervisor)

Professor HENG Pheng Ann (Committee Member)

Professor TANG Chi Keung (External Examiner)

Abstract

In this thesis, we propose a partially-blurred-image classification and analysis framework for automatically detecting images containing blurred regions and recognizing the blur types for those regions without performing blur kernel estimation and image deblurring.

Our method attempts to tackle two major problems. One is blur detection with simultaneous extraction of blurred regions. The result in this step provides useful high-level regional information, facilitating a variety of region-based image applications, such as content-based image retrieval, object-based image compression, video object extraction, image enhancement, and image segmentation. It can also serve as one of the criteria of measuring the quality of the captured images.

The second objective of our method is to automatically classify the detected blur regions into two types: near-isotropic blur (including out-of-focus blur) and directional motion blur. We classify image blur into these two classes because they are most commonly studied in the image restoration field. The classified blur images also easily find applications in motion analysis and image restoration.

We develop several blur features modeled by image color, gradient, and spectrum information, and use feature parameter training to robustly classify blurred images. In our system, blur detection and blur type classification are achieved in two steps. First, blurred images with blurry regions are detected from the given images. In this step, we make use of a combination of three features, namely, *Local Power Spectrum Slope*, *Gradient Histogram Span*, and *Maximum Saturation*. Second, directional motion blurred regions are distinguished from out-of-focus blurred

regions. We introduce another feature, i.e., *Local Autocorrelation Congruency*, in this step.

Our blur detection is based on image patches, making region-wise training and classification in a single image efficient. We extensively experiment with training and testing data with manually labelled patches, and use them to measure the accuracy of our patch classification algorithm. We also apply our algorithm to ranking the confidence of blur in the image level. Finally, our algorithm is applied to solving the problem of partially blurred image segmentation. Experiments show that our method works satisfactorily on challenging image data, which establishes a technical foundation for solving several computer vision problems, such as motion analysis and image restoration, using the blur information.

摘要

本論文中提出了一種局部模糊圖像的分析與分類系統。該系統能夠自動檢測含有模糊區域的圖像，并且可以在不需要進行模糊濾鏡的估計以及模糊消除處理的情況下判別該模糊區域的模糊類型。

本文的方法嘗試解決兩個主要問題。問題一是模糊檢測并同時提取模糊區域。其結果提供了有用的高階區域信息，有助于多種基于區域的圖像應用，例如基于內容的圖像檢索，基于目標的圖像壓縮，視頻目標提取，圖像增強以及圖像分割等等。同時，模糊檢測的結果還可以作為拍攝照片質量的衡量標準之一。我們的第二個問題是自動對檢測出的模糊區域根據模糊的類型分為兩類：近全向性模糊（包括失焦模糊）和方向性運動模糊。我們將圖像模糊分為這兩類因為它們是在圖像復原中最普遍的。這些分類后的模糊圖像也能夠很容易應用在移動分析以及圖像復原中。

我們通過對圖像的顏色，梯度和頻譜信息等方面建模，提出多種模糊特征，通過特征參數值的訓練學習來增強模糊圖像分類的魯棒性。在我們的系統中，模糊檢測與模糊類型分類是分兩步來實現的。首先，我們檢測出含有模糊區域的圖像。這一步中，我們結合利用了三種模糊特征，他們分別是局部功率譜斜率，梯度直方圖跨度，以及最大飽和度。第二步，我們進一步區分近全向性模糊和方向性運動模糊。這一步中，我們引入了另一個模糊特征，稱為局部自相關一致度。

我們對模糊的檢測是基于圖像的分片，這樣可以有效地進行一幅圖像內區域性的參數學習與分類。我們搭建了訓練數據庫與測試數據庫用

于大量的實驗測試，并人工對所有圖像分片標定類型，以測試我們對于分片分類的準確性。我們還將該算法應用于對于圖像模糊確信讀的排列。最后，我們將該算法應用于局部模糊圖像的分割。實驗顯示我們的算法在困難范例中都能得到較滿意的結果。利用提取的關于模糊的信息，可以為許多計算機視覺問題，例如運動分析及圖像復原等，提供一個技術的基礎平臺。

Acknowledgments

Foremost, I would like to thank my supervisor Prof. Jiaya Jia, whose support and guidance made my thesis work possible. He has closely worked with me and has always been available to advise me. I am very grateful for his patience, motivation, enthusiasm, and immense knowledge in Computer Vision that, taken together, make him a great mentor.

I want to thank present and past members of our lab : Wei Xiong, Leilei Xu, Qi Shan, Hin Shun Chung, Zhaorong Li, Jia'ning Chen, Li Xu and Chenjun Tao for giving me an introduction into the lab, sharing their knowledge, and answering my questions. Special thanks to Zhaorong Li for helping to finish up this project and cooperating in equation deduction and experiments in a tight schedule. Thanks to Xiaopei Liu for inspiring me the idea of using the feature of local autocorrelation.

I specially want to thank Prof. Heng Pheng Ann for his kind help and suggestions all the time during my study in the Chinese University of Hong Kong. And I also thank Prof. Wong Tien Tsin for all the instructive advices.

Finally, I thank my parents and families for their faith and support.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Images are produced to record or display useful information. However, in many cases, the recorded image represents a blurred or partially blurred version of the original scene. During previous a few decades, blur related topics have been investigated extensively in the field of computer vision and image processing, including image deblurring, blind deconvolution, depth from focus, etc.

Blurring is a form of bandwidth reduction of an ideal image caused by an imperfect image formation process [3, 4, 5, 6]. It can be due to relative motion between the camera and the original scene, or by an optical system that is out of focus. When arial photographs are produced for the remote sensing purposes, blurs are introduced by atmospheric turbulence, aberrations in optical system, and relative motion between the camera and the ground. Such blurring is not confined to optical images. For example, electron micrographs are corrupted by spherical aberrations of the electron lenses, and CT scans suffer from X-ray scatter.

In addition to these blurring effects, noise always corrupts any recorded image. Noise may be introduced by the medium through which the image is created (random absorption or scatter effects), by the recording medium (sensor noise), by measurement errors due to the limited accuracy of the recording system, and by quantization of the data for digital storage.

One of the popular classifications of blurring is based on the formation of the

image recording model. To be more specifically, the blur can be classified to *spatially invariant* and *spatially variant* types. *Spatially invariant* blur means that the blur is independent of the position. That is, a blurred object will look the same regardless of its position in the image. *Spatially variant* blur means that the blur depends on the position. That is, an object in an observed image may look different if its position is changed.

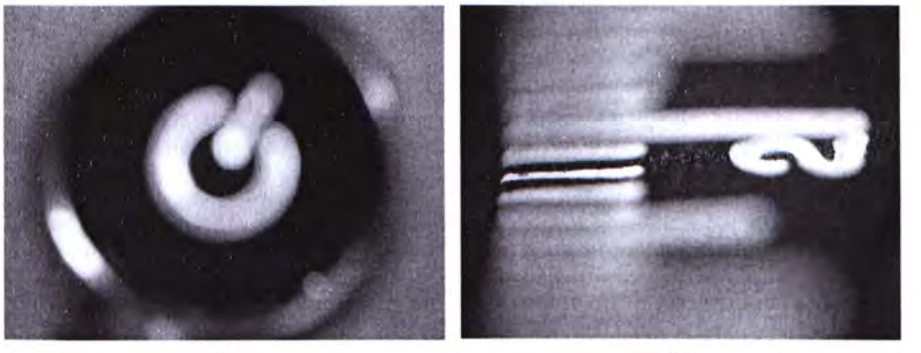
Another classification method is based on blur formation, and is normally classified into two types: motion blur and focal blur. For motion blurred images, there are mainly two types of motion blur exist. As shown in Fig. 1.1, the blur in the left picture is caused by the camera motion where the image is fully blurred by the camera vibration, while the right one is an example where the relative motion between the bus and the camera causes the blur and the background is still clear. Normally, the camera motion blur, without camera rotation, as shown in Fig. 1.1(a), can be regarded as a spatially invariant blur, whereas the object motion blur, as shown in Fig. 1.1(b) should be regarded as spatially variant blur.

(a) (b)

Figure 1.1: Two types of motion blur. (a) Camera motion blur, (b) Object motion blur.

On the other hand, for focal blurred images, there are also mainly two types of blurriness. Two examples are illustrated in Fig. 1.2. The left one is fully focal blurred, with none of the objects in the scene is clear. The auto-focus functionality of a digital camera is not always robust. Photographs taken with auto-focus setting

can still result in out-of-focus images. For some photos, such as that in Fig. 1.2(a), the entire image is out of focus, which may result from the complete failures of the auto-focus function when the object is too near to the camera lens. Fig. 1.2(b) is one example of low Depth of Field (DoF) images, which are generated through an important photographic technique commonly used to assist readers in understanding the depth information within a 2D photograph. For this kind of partially blurred photos, only objects with their distance between the camera falling into the range of depth of field will remain sharp, while all others will have out-of-focus blur in varying degrees.



(a) (b)

Figure 1.2: Two types of focal blurred images. (a) Fully focal blurred image, (b) Low Depth of Field image, with partially focal blurred regions.

In this thesis, we focus on detecting and analyzing partially blurred images, such as Fig. 1.1(b) and Fig. 1.2(b), and propose a novel method to automatically detect blurred images, extract possible blurred regions, and further classify them into two categories, i.e., near-isotropic blur and directional motion blur.

Our method attempts to tackle two major problems. One is blur detection with simultaneous extraction of blurred regions. The result in this step provides useful high-level regional information, facilitating a variety of region-based image applications, such as content-based image retrieval, object-based image compression, video object extraction, image enhancement, and image segmentation. It can also serve as one of the criteria of measuring the quality of captured images.

(a) (b)

Figure 1.3: Two image examples with (a) motion blurred regions and (b) out-of-focus blurred regions.

The second objective of our method is to automatically classify the detected blur regions into two types: near-isotropic blur (including out-of-focus blur) and directional motion blur. We classify image blur into these two classes because they are most commonly studied in image restoration. The blur classified images also easily find applications in motion analysis and image restoration. Two partial-blur-image examples are illustrated in Fig. 1.3.

Although topics of image blur analysis have attracted much attention in recent years, most previous work focuses on solving the deblurring problem. General blur detection, on the contrary, is seldom explored and is still far from practical. Rugna *et al.* [7] introduced a learning method to classify blurry or non-blurry regions in one input image. This method is based on an observation that blurry regions are more invariant to low pass filtering. In our experiments, we find that only using this information is not sufficient for designing a reliable classifier. Different blur measures should be combined in order to achieve high-quality blur detection. This method also does not distinguish blur types.

Utilizing the statistics of gradient information along different directions, the method in [1] builds an energy function based on the inferred blur kernel in order to segment image into blur/nonblur layers. This method only discovers motion blurred regions by inferring directional blur kernels. Other blur estimation methods, such

as the one proposed in [8], only provide a measure of blur extent, which cannot be directly used to discriminate blurry against non-blurry regions.

In this thesis, we present a new blur detection and analysis method for automatically extracting blurry regions by combining specifically designed blur features represented by spectral, gradient, and color information, respectively. Then we use the direction information of local autocorrelation function, which evaluates how well a local window in a blur region matches a spatially-shifted version of itself, to further discriminate the type of blur. In our blur classification, no blind deconvolution, which may involve complex kernel estimation, is performed.

Our features are extracted in local regions using the color-structure information relative to that collected globally in each image. This process is based on a key thought: if we directly use the image color information for each local region, it is hard to define a threshold for classification due to the variation of natural image structures. We, thus, take account of the relative information between the patch and the image, which makes our feature analysis reliable in regard to the color-structure diversity in natural images. Extensive experiments show that our method works satisfactorily with challenging image data.

Chapter 2

Related Work and System Overview

During the last few decades, blurred image related topics have been investigated intensely in the field of computer vision and image processing, such as image deblurring, blind deconvolution, depth from focus, etc. In this chapter, we first review related work of three most related topics, i.e. blur detection, image deblurring, and low Depth of Field (DoF) image auto-segmentation. Then an overview of our system is given.

2.1 Previous Work in Blur Analysis

We reviewe, in this chapter, three most related topics of our problem, i.e. blur detection, image deblurring, and low Depth of Field (DoF) image auto-segmentation. All of those topics deal with problems related with blurry photographs. So all of them need to analyze the formation, characteristics, and many aspects of blurry photos, and utilize special properties of blurry photos in order to tackle their problems. In the following, we will give some brief introduction of those methods.

2.1.1 Blur detection and estimation

In the topic of blur detection and estimation, previous approaches aim at measuring blur extent of edges and are based on the analysis of edge sharpness. They aim at measuring the sharpness or blurriness of edges in an image, which can be useful

for a number of applications in image processing, such as checking the focus of a camera lens, helping to identify shadows whose edges are often less sharp than object edges, the separation of variations in illumination from the reflectance of the objects in an image, known as intrinsic image extraction, and in-focus areas vs. out-of-focus areas in an image.

For example, a non-reference perceptual blur metric is presented in [9, 10], in order to measure the perceptural blur in an image or a video sequence. This metric is defined in the spatial domain as the spread of the edges, and blur is measured as the average edge width along vertical edges. Chuang *et al.* [11] extended this idea by first fitting gradient magnitude along edge direction to a normal distribution. Then the standard deviation of this distribution, together with gradient magnitude, are combined into a weighted average and regarded as the blur measure. The standard deviation describes the width of the edge, and the edge magnitude helps make the blur measure more reliable. The weight is calculated from the contrast of the input image and needs no manual inputs.

Elder and Zucker [8] proposed a method for edges detection and blur estimation, which modeled focal blur by a Gaussian blur kernel and calculated the response using the first and second order derivative steerable Gaussian basis filters [12]. Therefore, focal blur is estimated by the thickness of object contours. This method utilizes local scale control for local estimation at each point of the image, localizes edges over a broad range of blur scale and contrast and requires no input parameters other than the second moment of the sensor noise. It is proved to be useful in estimation of blur in complex images, where the smoothness assumptions underlying Fourier methods for blur estimation do not apply and the potential for interference between nearby edges of very different blur scale requires that estimates be made at the minimum reliable scale.

Zhang and Bergholm [13] defined Gaussian Difference Signature for multi-scale blur estimation and edge type classification in scene analysis. This signature functions similarly to the first-order derivative of Gaussian, in order to measure the

degree of diffuseness introduced by out-of-focus objects and classify edges into "diffuse" or "sharp". This classification is useful for scene understanding, provide a measure of depth, and can be used qualitatively for detecting occluding and occluded edge contours and segmentation purposes. After that, edges are further labeled with a profile type, including step edge, diffuse step edge, bar edge, diffuse bar edge, line edge and diffuse line edge. The classification results can contribute to better understanding of physical properties in the world.

Note that all these methods assume that the Point Spread Function (PSF) is modeled by a Gaussian blur filter. They cannot be applied to detecting ubiquitous non-Gaussian blur. However, we inspired from these method the correlation between edge sharpness, directional edge sharpness and the degree of blurriness.

2.1.2 Image deblurring

Normally, the process of image deblurring can be divided into two major steps. One is image restoration, the other is blind deconvolution. The field of image restoration (sometimes referred to as *image deconvolution*) is concerned with the reconstruction or estimation of the uncorrupted image from a blurred and noisy one. In order to perform the image deconvolution, the characteristics of the degrading system and the noise are assumed to be known as a priori. However, in practical situations, this information may not be able to obtain directly from the image formation process, which leads to another topic - blur identification. The goal of blur identification is to estimate the attributes of the imperfect imaging system from the observed degraded image itself prior to the restoration process. The combination of image restoration and blur identification is often referred to as *blind image deconvolution* [14]. Fig. 2.1 gives an illustration of the general steps involved in the process of image deblurring.

There have been various restoration filters proposed in the previous study of image deconvolution. Generally speaking, they can be classified into three classes

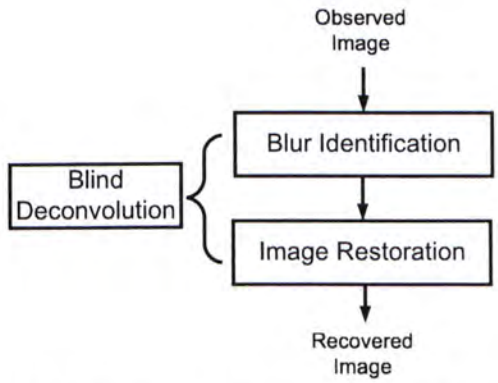


Figure 2.1: Flow chart of the image deblurring process

- inverse filter, least-square filter and iterative filter. An inverse filter is a linear filter whose point-spread function is the inverse of the blurring function. Since the existence of spectral zeros or small values of the original blurring function, the inverse filter is notoriously known to introduce excessively amplified noise.

Two commonly used least-square filters are Wiener Filter [15] and Constrained Least-Squares Filter [16]. Wiener Filter chooses the point-spread function to minimize a mean-squared error between the ideal and the restored image. This criterion makes the difference between the ideal image and the restored one, i.e. the remaining restoration error, as small as possible in average. Thus Wiener filter trades off the restoration by inverse filtering and suppression of noise for those frequencies where the frequency components of blur filter approach zero. However, that would introduce the "ringing" or "halo" artifacts near edges in the restored image. The formation of Constrained Least-Squares Filter is similar to that of the Wiener filter, except that it uses another more reasonable function taking account in a second smoothness constraint that minimize the high-frequency content of the restored image.

Iterative Filters [6, 17, 18] provide ways to handle the situations in which the dimension of the image to be restored is very large, or additional priori information is available about the restored image. Several optimization algorithms can be

used here, such as the steepest descent algorithm and the conjugate gradient algorithm. The most commonly used iterative non-blind deconvolution algorithm is the Richardson-Lucy(RL) algorithm [19].

The above methods were proposed to solve the classical linear image restoration problem, where the PSF is assumed to be known explicitly. Unfortunately, in many practical situations, the blur is often unknown, and little information is available about the true image. Therefore the true image must be identified directly from the observed image by using partial or no information about the blurring process and the true image. Such an estimation problem, assuming the linear degradation model, is called blind deconvolution.

There are several motivations behind the use of blind deconvolution for various image processing applications. It is often costly, dangerous, or physically impossible to obtain a priori information about the imaged scene. For example, in applications like remote sensing and astronomy, it is difficult to statistically model the original image or the specific information about scenes never imaged before [20, 21]. In addition, the degradation from blurring cannot be accurately specified. In real-time imaging, such as medical video-conferencing, the parameters of the PSF cannot be pre-determined to instantaneously deblur images [22]. So it is clear that classical image restoration methods that assume a known PSF are not suitable for tackling many image processing problems. In these cases, an algorithmic approach to combined blur identification and image restoration is required. In this sense, blind deconvolution is a practical way for image restoration. Indeed, existing research in the area [23, 24, 25] has shown its worth.

General blind deconvolution techniques incorporate partial information of the true image and PSF, e.g. statistical information, physical properties, etc, into an optimality criterion, which is minimized or maximized to find the estimates of the components (or their inverses).

The simplest blur identification procedure works as follows:

1. Choose a parametric blur model for the PSF.
    - The PSF can be a motion model or an out-of-focus model.
    - Model the PSF as a (small) set of coefficients within a given finite support. For instance, the blur PSF can be modeled as a square matrix of size 20 by 20.
2. Estimate the parameters in the chosen blur model.

There have been a large amount of blind deconvolution algorithms proposed. A possible classification of those existing algorithms is depicted in Fig. 2.2.

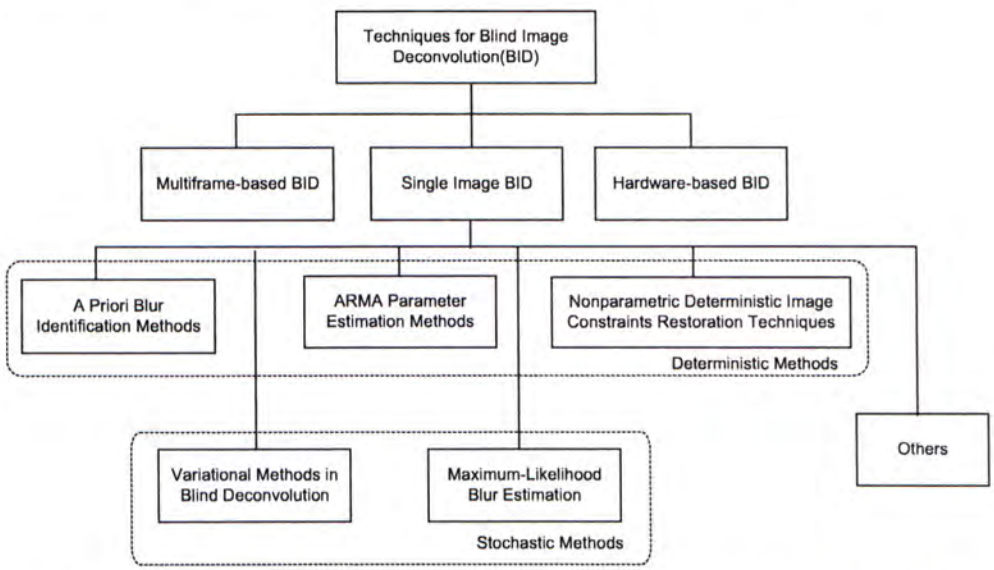


Figure 2.2: Classification of existing image blind deconvolution techniques.

In the following , several blur identification algorithms are briefly introduced with respect to the order in Fig.2.2. A priori blur identification methods [26, 27] perform the blind deconvolution by identifying the PSF prior for restoration. This kind of methods identify the PSF separately from the true image, in order to use it later with one of the known classical image restoration methods. These methods are based on the assumption that a parametric form of the PSF is required to be known, such as linear camera motion or an out-of-focus lens system. Then the

spectral zero pattern, as well as the blur model, could be estimated from the Fourier transform of the observed image. This can be viewed as a blur identification based on frequency domain zeros, which is one of the simplest method. An alternative way is to compute the 2D *cepstrum* of the observed image, and infer the blur parameters from the spikes of the cepstrum. After the PSF has been completely identified, one of the classical restoration techniques is used to estimate the true image.

ARMA parameter estimation methods modeled the true image as a 2D Auto-Regressive(AR) process and the PSF as a 2D Moving Average(MA) process. Then many methods are used to identifying the ARMA parameters, e.g. Maximum-likelihood(ML) estimation [28, 29, 30], Generalized Cross-Validation(GCV) [31], Neural Networks [32], and High Order Statistics(HOS) methods [33]. The AR model is valid for generally smooth and homogenous true images, but not valid for situations in which the true image has abrupt changes in local image characteristics, such as edges. Besides, using ML or GCV methods to estimate the parameters may converge to local minima.

Nonparametric deterministic image-based restoration techniques do not assume parametric models of either the image or the blur. They add deterministic constraints to true images, such as non-negativity, known finite support, and the existence of blur invariant edges, etc. These methods are iterative and simultaneously estimate the pixels of the true image and the PSF (or its inverse). The constraints on the true image and PSF are incorporated into an optimality criterion which is minimized using numerical techniques. Commonly used algorithms in this class include Iterative Blind Deconvolution (IBD) [34, 35], McCallum's Simulated annealing algorithm [36, 37], Aonnegativity and Support Constraints Recursive Inverse Filtering algorithm(NAS-RIF) [38, 39], and blind superresolution algorithms [40].

The blur deconvolution methods mentioned above mainly attempt to impose constraints on the image and the PSF in an alternating fashion. In other words, such approaches cycle between two (the image and the PSF) estimation steps. In the image estimation step, the image is estimated assuming that the PSF is fixed to its

last estimate from the PSF estimation step. In the PSF estimation step, the PSF is estimated assuming the image fixed to its last estimate from the image estimation step. Algorithms of this nature use a *deterministic framework* to introduce *a priori* knowledge in the form of convex sets, "classical" regularization, regularization with anisotropic diffusion functionals, and fuzzy soft constraints, etc. In the following, a probabilistic framework of maximum likelihood estimation using EM is reviewed.

In Maximum-Likelihood(ML) methods [41], the true image and additive noise of the imaging system are assumed to be white multivariate Gaussian processes, and are uncorrelated. Then, the problem of blind deconvolution becomes that of estimating the unknown parameters from the observed blurred image. So the Probability Density Function(PDF) of the blurred image conditioned on the unknown parameters is maximized to get the ML estimate of these parameters. Due to the high degree of nonlinearity, the optimization is difficult and is conducted by using the Expectation-Maximization(EM) algorithm.

Variational methods have been recently applied to blind deconvolution problems. Various distributions are proposed to approximate the probability distributions of the unknown image, blur, and unknown parameters. Named as ensemble learning, Miskin and MacKay [42] use gamma priors on the image and blur, and gamma distributions as hyperpriors for the unknown parameters of the priors. This model assumes that the image pixels are independent identically distributed, thus, does not capture the pixel correlation of natural images. Likas and Galatsanos [43] use normal distributions on the unknown blur and image and an improper constant prior for the unknown parameters. The problem with this hyperprior is that, the estimation process relies very sensitively on the amount of observational noise, as well as the initial estimates of the unknown parameters. Molina and Mateos [44] propose the use of Simultaneous Auto-Regressions(SAR) as prior distributions for the image and blur, and gamma distributions for the unknown parameters of the priors and the image formation noise. Fergus and Singh [45] adopted the basic idea

of Miskin and MacKay [42]. The blur is modeled as a mixture of exponential distributions and the unknown parameters(the noise variance) is formed as a gamma distribution. It is considered as one of the most successful techniques to reduce the blurring effect caused by camera shake.

Although recently many methods have been proposed in image deblurring, most of them only tackle spatially-invariant blur, i.e., all pixels in the input image are blurred by the same PSF. Some methods [1, 46, 47, 48] were proposed to tackle the partial blur problem with the help of user interaction or blur kernel assumption. For example, a transparency based MAP model is proposed in [46] aims to estimate the blur filter from the transparency information of the blurred object boundary. For all these methods, if the PSF can be correctly reconstructed, the type of blur is also known using the structure of PSF. However, in practice, blind deconvolution usually performs unsatisfactorily even by making restrictive assumptions on image and kernel structures. It does not handle well partially-blurred images. Besides, a visually plausible deconvolution result does not imply that the PSF is correctly estimated. These factors make blind deconvolution not a good choice for general blur detection in terms of efficiency and accuracy, especially for handling images in a large database.

2.1.3 Low DoF image auto-segmentation

Another type of blur analysis is low Depth of Field (DoF) image auto-segmentation. Low DoF is a photography technique which abstracts the photographer's intention by giving a clear focus only on an Object of Interest (OOI). An example of low DoF image is shown in Fig. 1.3(b). Many algorithms are proposed for unsupervised segmentation of low DoF images. The segmentation method in [49] calculates the pixel-wise spatial distribution of the high frequency components in the image and a thresholding method is applied thereafter. Similarly, morphological filters are used in [2] after extracting regions with large value of Higher Order Statistics (HOS).

In [50], localized blind deconvolution is applied to the image to produce a focus map, based on which low DoF images can be automatically classified into two types of regions: focused foreground or defocused background. In [51], a multi-scale context-dependent approach is proposed to segment images based on features extracted from wavelet coefficients in high frequency bands.

The afore-mentioned methods for automatic OOI extraction are not suitable for our blur detection because they only work on low DoF input images containing out-of-focus background. In [52], Low DoF images are detected by calculating a *low DoF indicator*, defined as the ratio of wavelet coefficients in high-frequency of the image central regions to that of the whole image. The higher the ratio is, the higher probability of low DoF this image is. This method simply assumes that low DoF images contain focused object near the center and surrounding pixels are out of focus. This method also does not suit our general-purpose blur detection.

Different from all previous work, in the following sections, we present an unified framework for blur detection and blur type classification. Our proposed blur detector makes use of information of spectral, gradient and color, automatically detects images with partially blurred regions, and further discriminates the types of blur inside those blurred regions.

2.2 System Overview

Due to the diversity of natural images, in this paper, we propose a learning framework with a training process to detect partially blurred images. These images are further classified into two blur types.

The basic flow chart of our system is given in Fig. 2.3. As shown Fig. 2.3, in our system, blur detection and blur type classification are achieved in two steps. After the feature extraction, detection of blurred images is first performed, followed by the step of blur type classification.

We also show an extended flow chart of our system in Fig. 2.4. In the step of

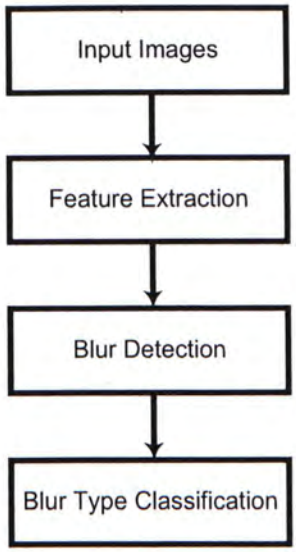


Figure 2.3: System basic flow chart

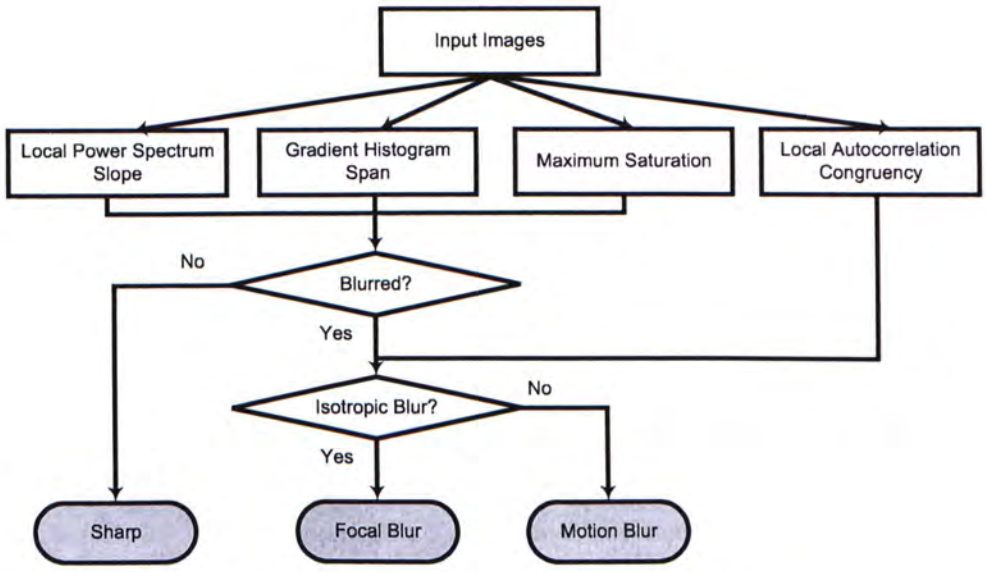


Figure 2.4: System extended flow chart

feature extraction, we extract four kinds of features, namely, *Local Power Spectrum Slope*, *Gradient Histogram Span*, *Maximum Saturation*, and *Local Autocorrelation Congruency*. After feature extraction, blur detection is performed. We propose using a combination of three features, i.e. Local Power Spectrum Slope, Gradient Histogram Span, and Maximum Saturation, to model the blur characteristics in different ways. Finally, directional motion blurred regions are distinguished from out-of-focus blurred regions by using another feature, i.e., Local Autocorrelation Congruency.

In order to handle partial blur, our approach adopts a region-based feature extraction. Specifically, we partition the input image into blocks and analyze features in each of them. We show in the following sections that this local representation provides reliable blur measures.

Note that automatic blur detection without really estimating the blur kernels is not straightforward. In our system, besides introducing spectral and visual clues, we also include analysis of the characteristics of the two blur types, which can provide useful clues for designing corresponding features. A detail analysis of the relationship between blur direction and edge direction is given in Appendix A.

Chapter 3

Blur Features and Classification

In this chapter, we detail our four blur features and show examples of how these features differentiate different blur types. Then the classification method is introduced.

3.1 Blur Features

There are four different features developed and combined in our system. These features are derived by analyzing the visual and spectral clues from images.

- *Local Power Spectrum Slope.* Due to the low-pass-filtering characteristic of a blurred region, some high frequency components are lost. So the amplitude spectrum slope of a blurred region tends to be steeper than that of an unblurred region.

- *Gradient Histogram Span.* The distribution of gradient magnitude serves as an important visual clue in blur detection. Blurred regions rarely contain sharp edges, which results in small gradient magnitude. Accordingly, the distributions of the log gradient magnitude for blurred regions should have shorter tails than that for other regions.

- *Maximum Saturation.* Unblurred regions are likely to have more vivid colors than blur regions. The maximum value of saturation in blurred regions is

correspondingly expected to be smaller than that in unblurred regions.

- *Local Autocorrelation Congruency.* If a region is blurred by relative motion between an object and the background in a certain direction, all edges of the object will be blurred, except those sharing the same direction with the motion. This is regarded as another important visual clue in our blur analysis.

A detail analysis of the relationship between the blur direction and edge direction is given in Appendix A. In our blur description, flat regions, i.e., blocks containing absolutely no edge structures, are ambiguous, since they can be interpreted as either blur or non-blur. So, in our system, these ambiguous regions are automatically removed beforehand. This makes our blur detection reliable for many natural images containing, for example, cloudless sky or textureless walls.

3.1.1 Local Power Spectrum Slope

We first compute the power spectrum of an image I with size $N \times N$ by taking the squared magnitude after Discrete Fourier transform (DFT)

$$S(u,v) = \frac{1}{N^2}|I(u,v)|^2, \tag{3.1}$$

where $I(u,v)$ denotes the Fourier transformed image. We then represent the two-dimensional frequency in polar coordinates, i.e., $u = f\cos\theta$ and $v = f\sin\theta$, and construct $S(f,\theta)$. According to [53, 54], by summing the power spectra S over all directions θ, $S(f)$, using polar coordinates, can be approximated by

$$S(f) = \sum_{\theta} S(f,\theta) \simeq A/f^{-\alpha}, \tag{3.2}$$

where A is an amplitude scaling factor for each orientation and α is the frequency exponent, called *slope of power spectrum.*

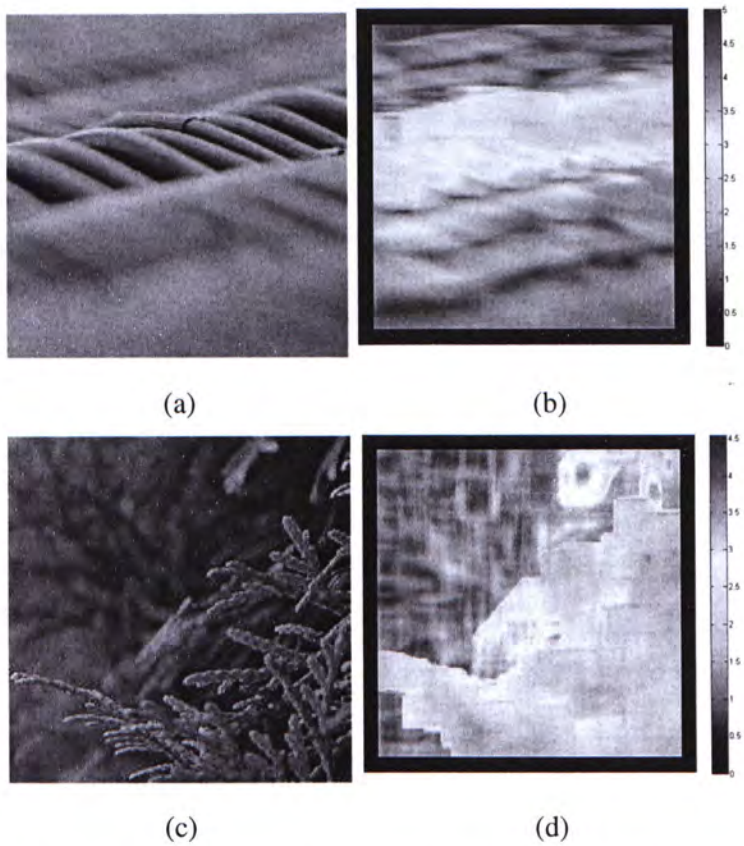


Figure 3.1: Local α_p illustration. α_p is calculated in patches with size 17×17 in these examples. (a) and (c) show two natural images. (b) and (d) illustrate the corresponding local α_p maps where warmer color pixels correspond to pixels with larger α_p values.

A number of studies [53, 54, 55] have demonstrated that, using image-wise computation, the power spectra of most natural images make $\alpha \approx 2$. A blurred image usually has a large α [56, 57] while an unblurred image, contrarily, corresponds to a small α. However, a natural image may contain objects with different boundary sharpness. Our experiments show that it is not reliable to simply set a threshold to α for blur estimation since the value of α varies in different images in a pretty wide range.

In [58], α discrimination experiments showed that human has sufficient ability

to discriminate changes in α even using very small image patches. This indicates that the blur is not determined by the absolute value of α over the whole image, but the relative local-to-global α differences.

Based on the above observation, we introduce the local and relative blur estimation using the slope of power spectrum α. We first compute α_o, the global measure of the slope of power spectrum for the whole image. Then we compare α_p computed in each local block p with α_o. If α_p is much larger than α_o, it is quite possible that this block is blurred. Our metric q_1, accordingly, is given by

$$q_1 = \frac{\alpha_p - \alpha_o}{\alpha_o}. \tag{3.3}$$

We show in Fig. 3.1 the computed local α_p with patch size 17×17. Pixels with warmer color indicate higher blur possibility. The α value distribution coincides with human visual perception.

3.1.2 Gradient Histogram Span

Recent research in natural image modeling [59] has shown that the gradient magnitudes of natural images usually follow a heavy-tailed distribution, which means natural images are with primarily small or zero gradient magnitudes. However, in a blurred image, since blurred regions usually do not contain too many sharp edges, the gradient magnitude distribution should have much of its mass with small or zero values.

For the two input images in Fig. 3.2(a) and (b), the gradient magnitude distributions for two blurred patches shown in Fig. 3.2(c) and (d) exhibit no apparent tail while those for two unblurred regions contain heavy tails as shown in Fig. 3.2(e) and (f).

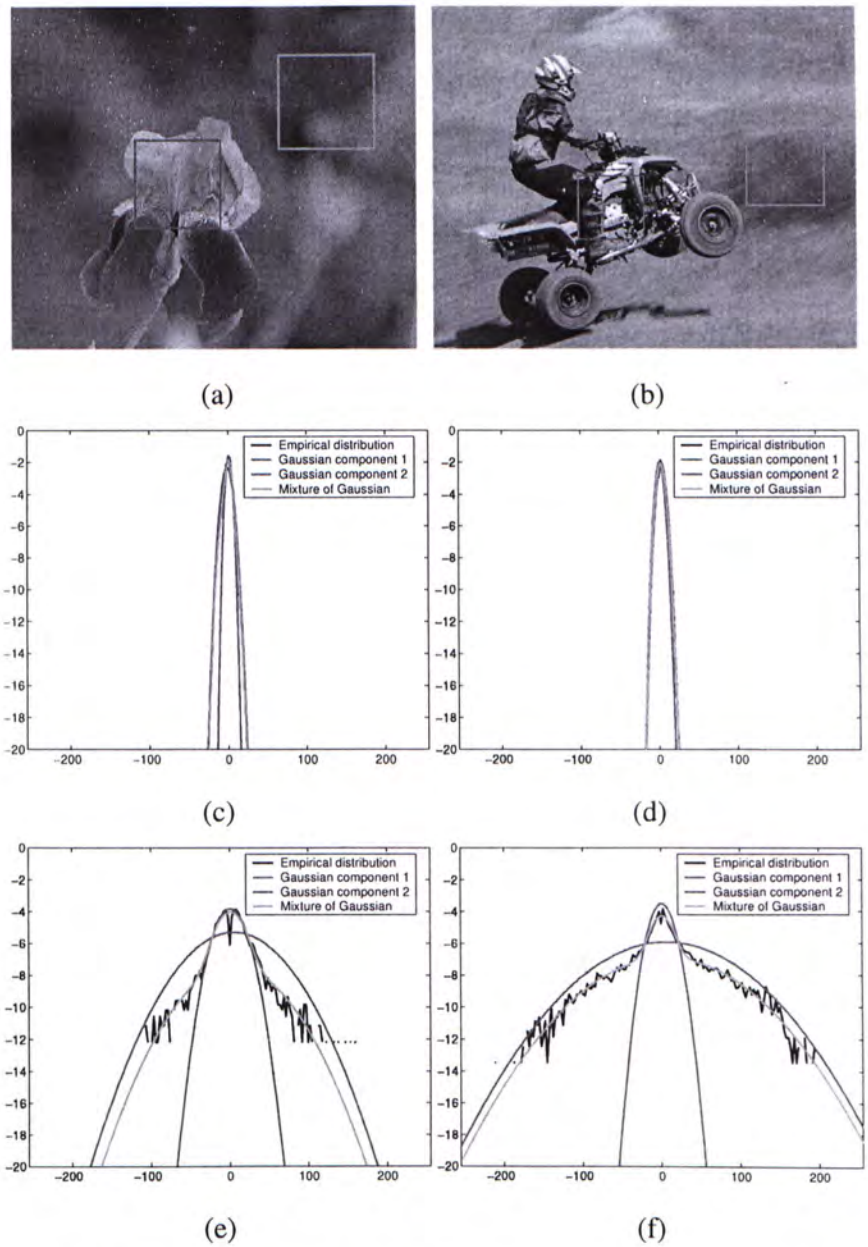


Figure 3.2: The distributions of gradient magnitudes and their approximations by mixture of Gaussians in patches. (a) One focal blurred image with two patches containing blurred and unblurred pixels respectively. (b) One motion blurred image with two patches selected similarly. (c) and (e) show the respective distributions of gradient magnitudes of the two patches in (a). The original distributions are shown in blue whereas the two Gaussian components are illustrated using red curves. (d) and (f) show the respective distributions of gradient magnitudes of the two regions in (b).

We choose to fit the local gradient magnitude distribution by a mixture of two-component Gaussian model using Expectation-Maximization:

$$\pi_0 G(x; u_0, \sigma_0) + \pi_1 G(x; u_1, \sigma_1) \tag{3.4}$$

with mean $u_0 = u_1 = 0$ and variance $\sigma_1 > \sigma_0$. The components of the fitted Gaussian mixture models are illustrated as red curves in Fig.3.2. The Gaussian component with larger variance σ_1 is mainly responsible for causing the heavy tail in the original distribution.

However, the value of σ_1 alone is not suitable to be used directly for blur detection. Different images containing unblurred objects may also have different levels of edge sharpness, making σ_1 variant. Thanks to our local patch blur detection scheme, the image containing the patch to be processed provides additional global information on measuring how the edges are formed and the gradient magnitude is distributed. So in computing blur gradient features, So we also adopt relative measurement and combine σ_1 with the structure contrast inside each patch. Specifically, we give large blur confidence to patches with significant structure contrast but relatively small gradient magnitude, which are most likely to be blurred. The local contrast is calculated by

$$C = \frac{L_{max} - L_{min}}{L_{max} + L_{min}}, \tag{3.5}$$

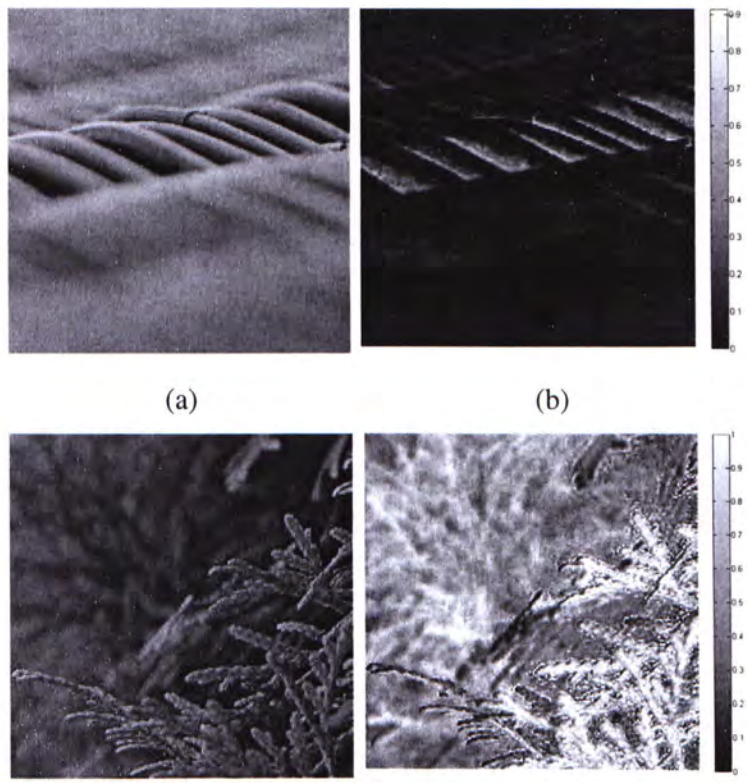
where L_{max} and L_{min} denote the maximum and minimum intensity values of all pixels in a region. We set the region size as 10×10 in our experiments, smaller than the size of a patch. Our metric q_2 for patch p is defined as

$$q_2 = \frac{\tau \sigma_1}{C_p + \epsilon^2}, \tag{3.6}$$

where C_p is calculated as the maximum local contrast C over all structural regions inside patch p, ϵ is a small value to prevent dividing by zero. Using our definition, q_2 will have small response when a patch has large C_p but small σ_1. Parameter τ is set to 25 in experiments to balance the scales of σ_1 and C_p.

3.1.3 Maximum Saturation

Beside the information we can get from the photo illuminance, we also analyzed the color components. It is observed that blurred pixels tend to have less vivid colors than unblurred pixels because of the smoothing effect of the blurring process. The illustration of image saturation is shown in Fig.3.3. As we can see that higher saturation values are mainly distributed in sharper regions.



(a) (b)

(c) (d)

Figure 3.3: Illustration of color saturation of images. (b) and (d) are the saturation map of color images of (a) and (b), respectively. We can see that higher saturation values are mainly distributed in sharper regions.

So we first compute pixel saturation by

$$S_p = 1 - \frac{3}{(R+G+B)} \left[\min(R,G,B)\right]. \tag{3.7}$$

Then, within each patch p, we compute saturation S_p for each pixel and find the maximum value $\max(S_p)$. It will be compared with $\max(S_o)$, the maximum saturation value of the whole image. Our metric q_3 is defined as

$$q_3 = \frac{\max(S_p) - \max(S_o)}{\max(S_o)}. \tag{3.8}$$

Noted that, for natural images, $\max(S_o)$ should be always larger than zero.

3.1.4 Local Autocorrelation Congruency

The image power spectrum is related to the autocorrelation function through the Wiener-Khinchin theorem, which states that the auto-correlation function and the power spectrum form a Fourier transform pair [60]. So besides the analysis of local power spectrum slope in defining feature q_1, we further analyze the local autocorrelation characteristics of blurred regions.

The local autocorrelation function [60] is a measure of how well a signal matches a time-shifted version of itself. As for an image, it can be interpreted as a measure of how well a local window in a blurred region matches a spatially-shifted version of itself. In our method, we use the directional information of local autocorrelation function to further discriminate the type of blurs inside those detected blur regions.

The local autocorrelation function at a point (x, y) under a shift $(\Delta x, \Delta y)$ in image space can be calculated by

$$f(x, y) = \sum_{(x_k, y_k) \in W} \left[I(x_k, y_k) - I(x_k + \Delta x, y_k + \Delta y) \right]^2,$$

where W is a local window centered at point (x, y). Using first-order Taylor expansion to approximate $I(x_k + \Delta x, y_k + \Delta y)$, $f(x, y)$ can be represented using a local autocorrelation matrix M:

$$\begin{aligned} f(x, y) &\simeq \sum_{(x_k, y_k) \in W} \left\{ \left[I_x(x_k, y_k), I_y(x_k, y_k) \right] \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} \right\}^2 \\ &= \left[\Delta x, \Delta y \right] M \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix}, \quad \text{where} \end{aligned}$$

$$M = \sum_{(x_k, y_k) \in W} \begin{bmatrix} I_x^2(x_k, y_k) & I_x(x_k, y_k) I_y(x_k, y_k) \\ I_x(x_k, y_k) I_y(x_k, y_k) & I_y^2(x_k, y_k) \end{bmatrix}.$$

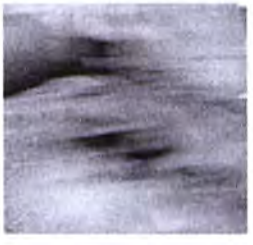
When taking a close look at the blurring process, one can notice that the color of one pixel is spread to its neighborhood after blurring, increasing its color similarity to its neighboring pixels. Besides, the color spreading strategy of each pixel is defined by the PSF. So the local autocorrelation function in blurred images, which depends on $(\Delta x, \Delta y)$, should also be strongly related to the shape and value of PSF. A detail analysis of the relationship between the blur direction and edge direction is given in Appendix A.

In fact, the Harris corner descriptor [61] is closely related to local autocorrelation function by adding a window function when calculating matrix M. If the window function is binary and rectangular, the two representations are actually identical. So we adopt the idea of Harris corner detector and calculate matrix M with a Gaussian window function to suppress noise.

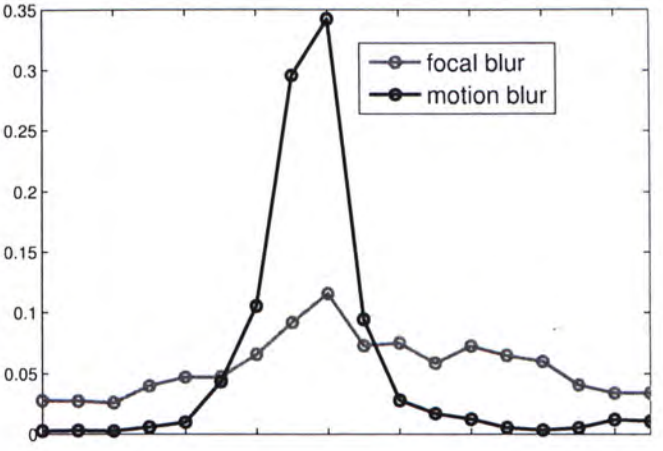
Using Cauchy-Schwarz inequality, we can prove that no matter what shape or distribution the window function has, the determinant of matrix M is larger than or equal to zero, i.e. $\text{Det}(M) \geq 0$. Equality holds if and only if $I_x(x_k, y_k)$ and $I_y(x_k, y_k)$ are linearly dependent for all $(x_k, y_k) \in W$, i.e. $I_y(x_k, y_k) = \beta I_x(x_k, y_k)$. Noted that, scalar β actually represents the gradient direction of point (x_k, y_k). So this equality condition requires exactly the same gradient direction for all points in a local window, which is rarely satisfied for natural images. We could expect that $\text{Det}(M) > 0$ always holds, so the contour of local autocorrelation function is ellipse-shaped.

We compute the eigenvalues $\lambda_1(x, y)$ and $\lambda_2(x, y)$ of M for each pixel (x, y) inside an image patch p, where $\lambda_1(x, y) > \lambda_2(x, y)$, and denote their corresponding eigenvectors as $V_1(x, y)$ and $V_2(x, y)$. The values of the two eigenvalues are proportional to the principal curvatures of the local autocorrelation function and form a rotationally invariant description of Matrix M.

(a)



(b)



(c)

Figure 3.4: The histograms of weighted eigenvector directions for two sample patches. (a) A focal blurred patch. (b) A motion blurred patch. (c) Histograms of weighted eigenvector directions of (a) and (b). The blue histogram is for the focal blurred patch and the red histogram is for the motion blurred patch. Directional motion blur produces distinctive peak value.

To be more specific, the eigenvector $V_2(x, y)$, which corresponds to the smaller eigenvalue $\lambda_2(x, y)$, represents the major axis direction of the ellipse-shaped contour of a local autocorrelation function at point (x, y). We thus construct a directional response histogram $hist(\theta)$ for patch p, where each bin represents one direction θ and the value of each bin is the number of pixels with eigenvector V_2 along direction θ in patch p, weighted by their ellipse axis ratio $\sqrt{\lambda_1/\lambda_2}$. One illustration is shown in Figure 3.4. In order to use the information of the histogram, we also normalize the bin values by dividing the total pixel number in one patch. Then our Local Autocorrelation Congruency measure is to check whether the histogram in this region has a distinctive peak. In our method, we compute the variance of the normalized bin values as our feature of Local Autocorrelation Congruency, i.e.

$$q_4 = \mathrm{Var}\{hist(\theta)\}. \tag{3.9}$$

3.2 Classification

Using the defined blur features, our system consists of two steps. In the first step, we train a blur/nonblur classifier to discriminate different regions. This classifier is based on features proposed in Sections 3.1.1, 3.1.2, and 3.1.3. Then, in the second step, the detected blurry regions are measured by Local Autocorrelation Congruency in order to recognize the blur types.

Given the list of features for discriminating unblurred and blurred regions, we apply the Bayes classifier:

$$\begin{aligned} \eta_a &= \frac{P\left(Blur \mid q_1, q_2, q_3\right)}{P\left(Sharp \mid q_1, q_2, q_3\right)} \\ &= \frac{P\left(q_1, q_2, q_3 \mid Blur\right) P\left(Blur\right)}{P\left(q_1, q_2, q_3 \mid Sharp\right) P\left(Sharp\right)}, \end{aligned} \tag{3.10}$$

where $P\left(Sharp \mid q_1, q_2, q_3\right)$, by the Bayes' rule, denotes the probability of labeling a region as "nonblur" given the defined features. Assuming the independence of different features, we simplify (3.10) to

$$\eta_a = \frac{P\left(q_1 \mid Blur\right) ... P\left(q_3 \mid Blur\right)}{P\left(q_1 \mid Sharp\right) ... P\left(q_3 \mid Sharp\right)}. \tag{3.11}$$

Then each conditional probability above is trained using the training image set.

In the second step, we further classify blurred regions into "motion blur" and "focal blur", using our feature defined in Section 3.1.4. Conditional probabilities $P\left(q_4 \mid focal\, blur\right)$ and $P\left(q_4 \mid motion\, blur\right)$ are trained, and the blur type classification is achieved by computing $\eta_b = P\left(q_4 \mid focal\, blur\right) / P\left(q_4 \mid motion\, blur\right)$.

Chapter 4

Experiments and Results

We describe our experiments in this section. In the first part, we construct training and testing datasets with manually labeled patches, and use them to test the accuracy of our patch classification algorithm. We also apply our algorithm to ranking the confidence of blur for an image. Finally, we experiment with partially blurred image segmentation.

4.1 Blur Patch Detection

We collect totally 100 partially blurred images and 100 unblurred images from photo sharing websites, such as Flickr.com and PBase.com, to form our dataset. In each category, half of the images are used for training and the other half are for testing. All the images are manually segmented into square patches. The size of each patch ranges from 50×50 to 500×500 pixels, which occupies $5\% \sim 20\%$ of the size of the original images. Examples of images and patches in our datasets are shown in Fig. 4.1. Then we label each patch as one of the following three types: "sharp", "motion blur", or "focal blur". In total, we generated 290 focal blur patches, 217 motion blur patches, and 516 unblurred patches from training set and 223, 139, and 271 patches, respectively, from testing set.

We evaluate the performance of our classifier using the Precision-Recall curve. Let N be the number of patches to be classified, f_i be the label for patch i, and a_i

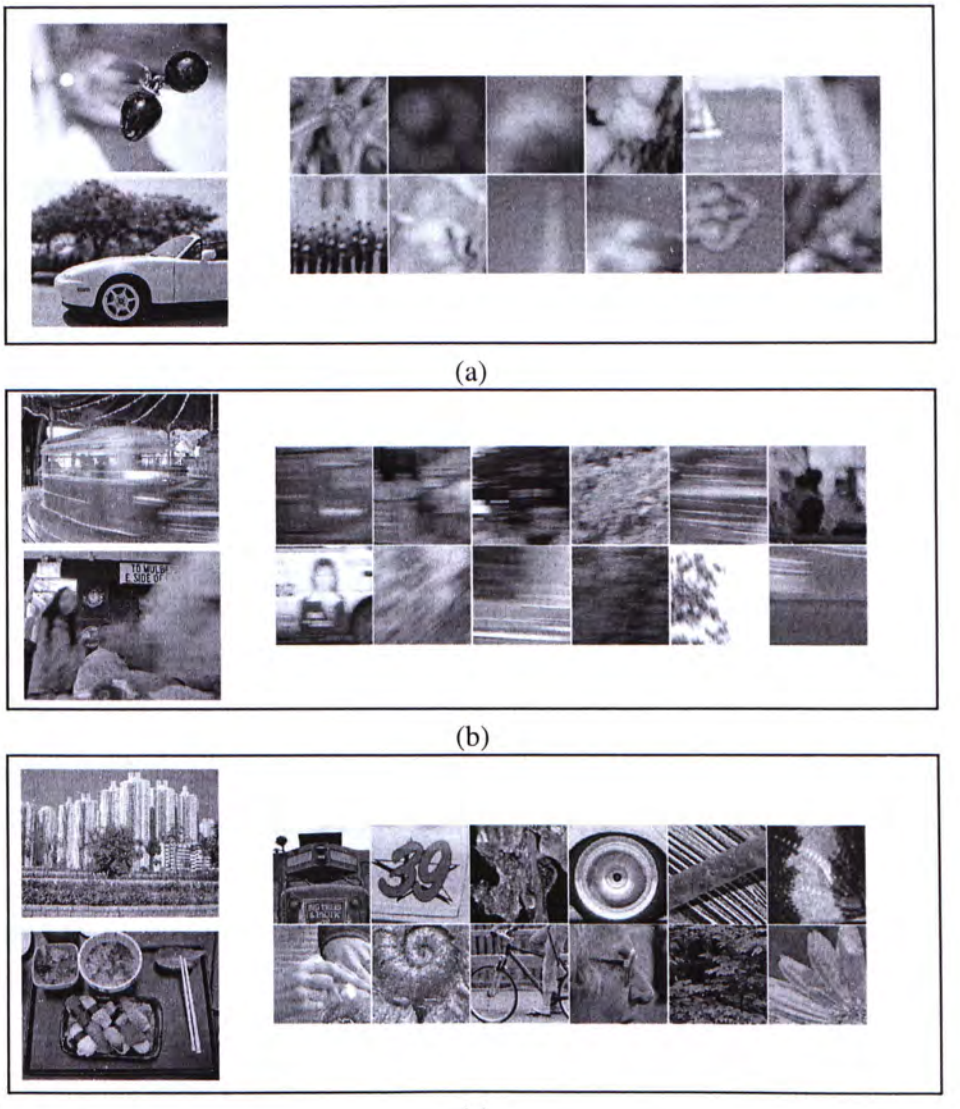


Figure 4.1: Selected examples of images (left) and manually labeled patches (right) from our datasets. (a) Partially focal blurred images and "focal blur" patches. (b) Partially motion blurred images and "motion blur" patches. (c) Sharp images and "sharp" patches.

be the ground truth label for patch i, we define the measurements as

$$\begin{aligned} Recall &= \frac{|\{i; f_i = a_i \& a_i = true\}|}{|\{i; f_i = ture\}|}, \\ Precision &= \frac{|\{i; f_i = a_i \& a_i = true\}|}{|\{i; a_i = true\}|}, \\ Accuracy &= \frac{|\{i; f_i = a_i\}|}{N}. \end{aligned}$$

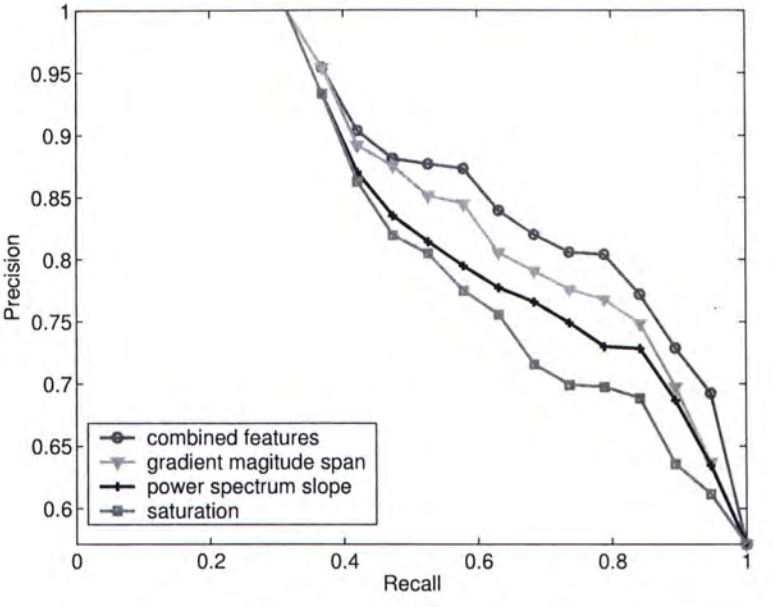
In the first step of our algorithm, patches labeled as "blur" are considered as $true$ instances in evaluating blur/nonblur classifier whereas, in the second step, patches labeled as "motion" are considered as $true$ in evaluating motion/focal blur classifier.

For evaluating individual features in blur detection, we plot precision-recall curves to show the discriminatory power. Similar to the definition of η_a, we set $\eta_i = P(q_i \mid Blur) / P(q_i \mid Sharp)$, where $i = \{1, 2, 3\}$ and show in Fig. 4.2(a) the precision-recall curves for each blur metric η_i and the combined metric η_a in classifying blur/nonblur regions. Our classifier η_a, in general, is with the best performance and achieves over 95% precision for low recall. Among our blur metrics, the feature of gradient magnitude span η_2 shown in blue is the most discriminative one. This is not surprising since blurred patches rarely contain steep changes of intensities. Table 4.1 shows the maximum accuracy rate, as well as the overall accuracy rate calculated by averaging over all recall levels. The maximum accuracy rate 76.98% for blur/nonblur classification is achieved when $\eta_a = 0.4$. This threshold is used later for our application of blur region recognition and segmentation.

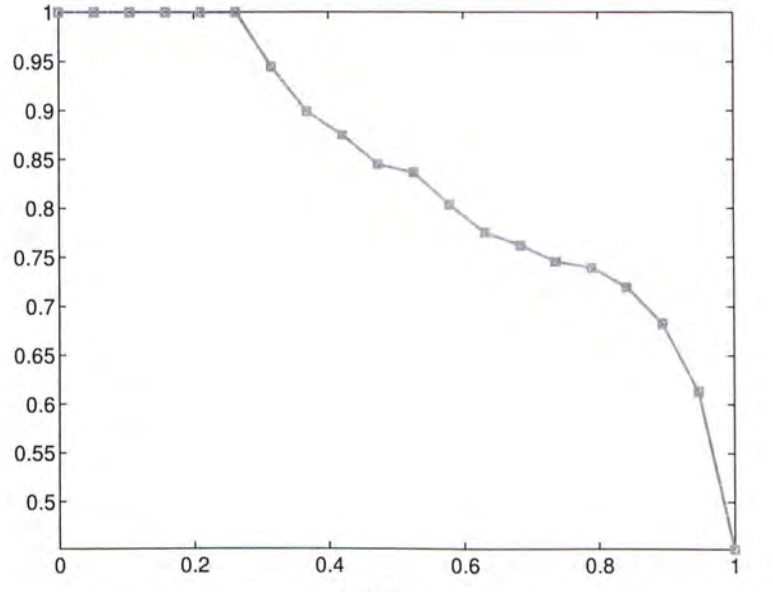
Classification Tasks	Overall Accuracy Rate	Maximum Accuracy Rate
Blur / Nonblur	63.78%	76.98%
Motion / Focal blur	65.45%	78.84%

Table 4.1: Accuracy Rate on the testing dataset.

In the second step, we test motion/focal blur classification using η_b. Fig. 4.2(b) shows the precision-recall curve. As listed in Table 4.1, the maximum accuracy rate is 78.84% when $\eta_b = 1.3$. This threshold is also used later for blur segmentation.



(a)



(b)

Figure 4.2: Precision-Recall curves of our classification. (a) Blur/nonblur classification results. The curve corresponding to the classifier using η_a is shown in red, whereas the curves for classifiers using η_1, η_2, and η_3, are shown in green, blue, and purple, respectively. (b) Precision-Recall curve for motion/focal blur classification.

Because of the similarity of the blurred and low-contrast regions in natural images, our classification results inevitably contain errors. We examined incorrectly classified patches and found that the latent ambiguous texture or structure in patches is the main cause of errors. For example, Fig. 4.3(a) and (b) show patches wrongly classified as blurred regions due to the shadow or low contrast pixels. The patch in Fig. 4.3(c) is mistakenly classified as motion blur because of the strong directional structures.

(a) (b) (c)

Figure 4.3: Examples of wrongly classified patches.

4.2 Blur degree

We introduce a method to measure the degree of blurriness of each images in this section. Our measurement and ranking of the degree of image blur is based on the same category of photographs. The patch-based blur detection and classification can serve as a foundation for ranking the degree of image blur. In experiments, we collect a set of flower pictures searched from Google and Flickr. Each image is segmented into patches with size 20×20. We rank each image by a blur confidence value defined proportional to the number of blurred regions in each image. Fig. 4.4 shows the blur recognition results by our method where the blur confidence of each image increases from left to right, coincident with human visual perception.

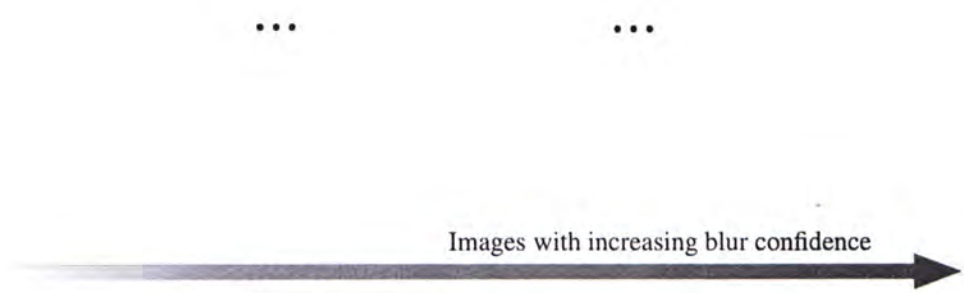


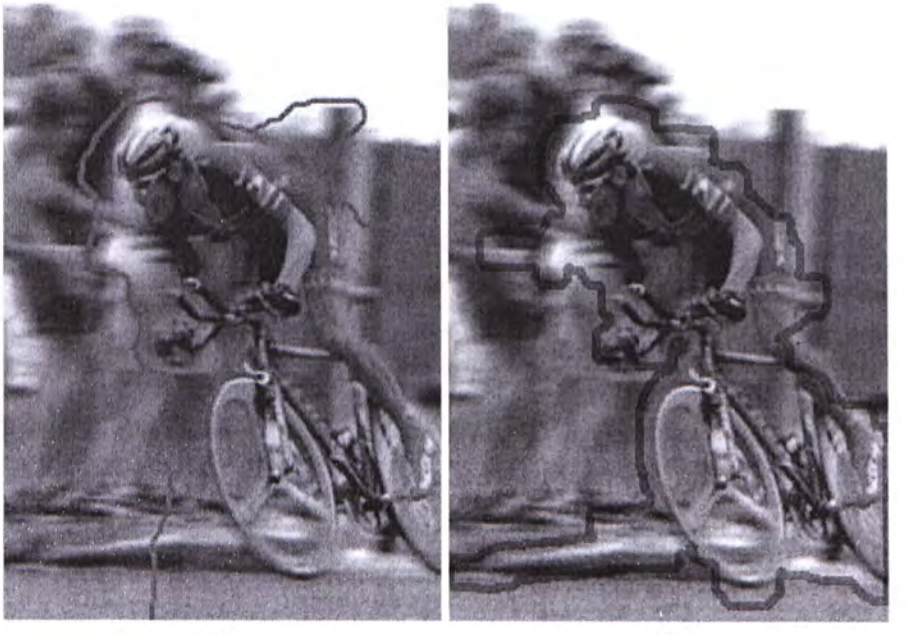
Figure 4.4: Partial blur recognition for flower images. Images are shown in an ascending order in terms of the size of blurry regions.

4.3 Blur Region Segmentation

Our method can also be used in partially blurred image segmentation. Because the blur classification is performed on patches, we detect the blur information of one pixel by using a local patch centered at it. The spatial similarity of blur types for neighboring pixels is also considered by introducing a smoothness term similar to that defined in [62]. Combining the blur detection and the pair-wise blur type similarity, we construct a Markov Random Field for each image. The corresponding Gibbs energy is minimized by applying the Graph Cuts method [63].

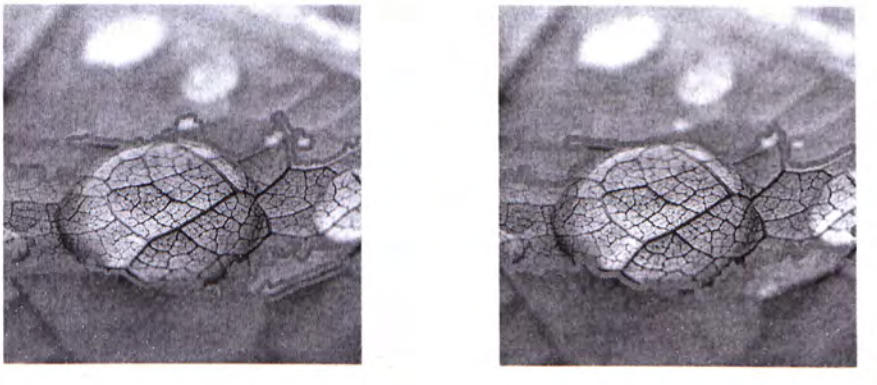
We show one blur segmentation example of partially motion blurred image in Fig. 4.5, where (a) and (b) give a comparison of the segmentation results produced by the method described in [1] and our approach. Fig. 4.5(a) is acquired directly from paper [1]. With comparable segmentation results, our method do not need to infer the blur kernel and make the directional motion assumption. Our features are computed within patches of size 50×50 centered at each pixel.

Another example of our segmentation results of low DoF images is shown in Fig. 4.6(b). We compare it to the segmentation result using method in [2], shown in



(a) (b)

Figure 4.5: Comparison of blur segmentation results for partially motion blurred images. (a) Blur segmentation result presented in [1]. (b) Our result on the same example without needing to infer the blur kernels.



(a) (b)

Figure 4.6: Comparison of segmentation results for partial-blurred low DoF images. (a) The segmentation result of a low DoF image using method in [2]. (b) Our result with more smooth segmentation boundaries.

(a) (b)

Figure 4.7: Blur segmentation results for partially blurred images. (a) shows a challenging image example containing unblurred, motion blurred, and focal blurred regions. (d) Our segmentation result with unblurred regions in red, motion blurred region in yellow, and focal blurred regions in blue.

Fig. 4.6(a). This method uses higher order statistics, i.e. the fourth-order moment of pixel variations in neighborhood, combined with morphological filters and adaptive thresholds to segment object of interest. Our segmentation result shows comparable performance in this case and more smooth segmentation boundaries. However, one thing to be clarified is that, since all the blur features we proposed are region-based, the precision level of our blur layer segmentation is highly depended on the patch size we used to calculated our features. In the example of Fig. 4.6, we use patches of size 10×10 to calculate all the features.

Finally, we tried our method for the challenge work of segmenting images containing both of the two types of blur. Fig. 4.7(a) is a challenge image example containing unblurred, motion blurred, and focal blurred regions. We show in Fig. 4.7(b) that our method is able to partition an image into several segments with respect to different blur types. In this example, we apply our blur/nonblur classifier to first segment out the unblurred regions (shown in red in (b)) and then extract motion blurred regions (shown in yellow in (b)). Differently blurred regions are faithfully

segmented, indicating that our automatic method can serve as a reliable basis for many region-based image applications, such as motion estimation, content-based image retrieval, and object extraction.

Chapter 5

Conclusion and Future Work

In this thesis, we have proposed a partial-blur image detection and analysis framework for automatically classifying whether one image contains blurred regions and what types of blur occur without needing to performing image deblurring.

Several blur features, measuring image color, gradient, and spectrum information, are utilized in a parameter training process in order to robustly classify blurred images. Extensive experiments show that our method works satisfactorily with challenging image data and can be applied to partial-blur image detection and blur segmentation. Our method, in principal, provides a foundation for solving many blur-oriented and region-based computer vision problems, such as content-based image retrieval, image enhancement, high-level image segmentation, and object extraction.

Moreover, our features are extracted in local regions using the color-structure information relative to that collected globally in each image. We take account of the relative information between the patch and the image, which makes our feature analysis reliable in regard to the color-structure diversity in natural images.

We construct extensive experiments with training and testing datasets with manually labelled patches, and use them to test the accuracy of our patch classification algorithm. We also apply our algorithm to ranking the confidence of blur for an image. Finally, our algorithm is applied to solve the problem of partially blurred image segmentation.

However, due to the similarity of the blurred and low-contrast regions in natural images, our classification results inevitably contain errors. So how to make our system work more robust is the main problem of future work. Other future works involve further utilizing the blur features to segment images into layers with different blur extent, as well as applying different classification methodologies to make our classification results more effective and robust.

Bibliography

[1] A. Levin, Blind motion deblurring using image statistics, in *NIPS*, pages 841–848, 2006.

[2] C. Kim, *Segmenting a Low-Depth-of-Field Image Using Morphological Filters and Region Merging*, IEEE Transactions on Image Processing **14**, 1503 (2005).

[3] A. Bovik and J. Gibson, *Handbook of Image and Video Processing*, Academic Press, Inc. Orlando, FL, USA, 2000.

[4] M. Banham and A. Katsaggelos, *Digital image restoration*, IEEE Signal Processing Magazine **14**, 24 (1997).

[5] R. Lagendijk, A. Katsaggelos, and J. Biemond, *Iterative identification and restoration of images*, International Conference on Acoustics, Speech, and Signal Processing , 992 (1988).

[6] J. Biemond, R. Lagendijk, and R. Mersereau, *Iterative methods for image deblurring*, Proceedings of the IEEE **78**, 856 (1990).

[7] J. Da Rugna and H. Konik, Automatic blur detection for metadata extraction in content-based retrieval context, in *SPIE*, volume 5304, pages 285–294, 2003.

[8] J. H. Elder and S. W. Zucker, *Local Scale Control for Edge Detection and Blur Estimation*, PAMI **20**, 699 (1998).

[9] P. Marziliano, F. Dufaux, S. Winkler, and T. Ebrahimi, A no-reference perceptual blur metric, in *ICIP (3)*, pages 57–60, 2002.

[10] P. Marziliano, F. Dufaux, S. Winkler, and T. Ebrahimi, *Perceptual blur and ringing metrics: application to JPEG2000*, Signal Processing: Image Communication **19**, 163 (2004).

[11] Y. Chung, J. Wang, R. Bailey, S. Chen, and S. Chang, *A non-parametric blur measure based on edge analysis for image processing applications*, IEEE Conference on Cybernetics and Intelligent Systems **1** (2004).

[12] W. T. Freeman and E. H. Adelson, *The Design and Use of Steerable Filters*, PAMI **13**, 891 (1991).

[13] W. Zhang and F. Bergholm, *Multi-Scale Blur Estimation and Edge Type Classification for Scene Analysis*, International Journal of Computer Vision **24**, 219 (1997).

[14] D. Kundur and D. Hatzinakos, *Blind image deconvolution*, IEEE Signal Processing Magazine **13**, 43 (1996).

[15] R. Gonzalez and R. Woods, *Digital Image Processing*, Prentice Hall, 2007.

[16] B. Hunt, *The application of constrained least squares estimation to image restoration by digital computer*, IEEE Transactions on Computers **22**, 805 (1973).

[17] A. katsaggelos, *Iterative image restoration algorithms*, Optical Engineering **28**, 735 (1989).

[18] J. Biemond and R. Lagendijk, *Iterative Identification and Restoration of Images*, Springer, 1991.

[19] W. Richardson, *Bayesian-based iterative method of image restoration*, J. Opt. Soc. Am **62**, 55 (1972).

[20] R. Bates, *Astronomical speckle imaging*, Physics Reports **90**, 203 (1982).

[21] J. Muller, *Digital Image Processing in Remote Sensing*, Taylor & Francis, 1988.

[22] M. Crowther and M. Fernandez, *Imaging in videoconferencing now: telemedicine in Somalia*, Advanced Imaging , 28 (1993).

[23] V. Krishnamurthi, Y. Liu, T. Holmes, B. Roysam, and J. Turner, *Blind deconvolution of 2D and 3D fluorescent micrographs*, Proc. SPIE **1660**, 95 (1992).

[24] T. Schulz, *Multiframe blind deconvolution of astronomical images*, J. Opt. Soc. Am. A **10**, 1064 (1993).

[25] A. Qureshi and H. Mouftah, *Partially-blind image restoration using constrained Kalmanfiltering*, International Conference on Acoustics, Speech, and Signal Processing , 3713 (1991).

[26] M. Cannon, *Blind deconvolution of spatially invariant image blurs with phase*, IEEE Transactions on Acoustics, Speech, and Signal Processing **24**, 58 (1976).

[27] A. Erdem and A. Tekalp, *Blur identification using bispectrum*, International Conference on Acoustics, Speech, and Signal Processing , 1961 (1990).

[28] A. Tekalp, H. Kaufman, and J. Woods, *Identification of image and blur parameters for the restoration of noncausal blurs*, IEEE Transactions on Acoustics, Speech, and Signal Processing **34**, 963 (1986).

[29] R. Lagendijk, A. Tekalp, and J. Biemond, *Maximum Likelihood Image and Blur Identification: A Unifying Approach*, Opt. Eng. **29**, 422 (1990).

[30] R. Lagendijk, J. Biemond, and D. Boekee, *Hierarchical blur identification*, International Conference on Acoustics, Speech, and Signal Processing , 1889 (1990).

[31] S. Reeves and R. Mersereau, *Blur identification by the method of generalized cross-validation*, IEEE Transactions on Image Processing **1**, 301 (1992).

[32] C. Cho and H. Don, *Blur identification and image restoration using a multilayer neuralnetwork*, IEEE International Joint Conference on Neural Networks , 2558 (1991).

[33] C. Nikias and A. Petropulu, *Higher-order spectra analysis: a nonlinear signal processing framework*, PTR Prentice Hall, 1993.

[34] G. Ayers and J. Dainty, *Iterative blind deconvolution method and its applications*, Optics Letters **13**, 547 (1988).

[35] F. Tsumuraya, N. Miura, and N. Baba, *Iterative blind deconvolution method using Lucy's algorithm*, Astronomy and Astrophysics **282**, 699 (1994).

[36] B. McCallum, *Blind deconvolution by simulated annealing*, Optics Communications **75**, 101 (1990).

[37] S. Kirkpatrick, C. Gelatt Jr, and M. Vecchi, *Optimization by Simulated Annealing*, Science **220**, 671 (1983).

[38] D. Kundur, *Blind Deconvolution of Still Images using Recursive Inverse Filtering*, PhD thesis, University of Toronto, 1995.

[39] D. Kundur and D. Hatzinakos, *A novel blind deconvolution scheme for image restoration usingrecursive filtering*, IEEE Transactions on Signal Processing **46**, 375 (1998).

[40] K. Nishi and S. Ando, *Blind superresolving image recovery from blur-invariant edges*, IEEE International Conference on Acoustics, Speech, and Signal Processing (1994).

[41] A. Katsaggelos and K. Lay, *Maximum likelihood blur identification and image restoration usingthe EM algorithm*, IEEE Transactions on Signal Processing **39**, 729 (1991).

[42] J. Miskin and D. MacKay, *Ensemble learning for blind image separation and deconvolution*, Advances in Independent Component Analysis , 123 (2000).

[43] A. Likas and N. Galatsanos, *A variational approach for Bayesian blind image deconvolution*, IEEE Transactions on Signal Processing **52**, 2222 (2004).

[44] R. Molina, J. Mateos, and A. Katsaggelos, *Blind Deconvolution Using a Variational Approach to Parameter, Image, and Blur Estimation*, IEEE Transactions on Image Processing **15**, 3715 (2006).

[45] R. Fergus, B. Singh, A. Hertzmann, S. T. Roweis, and W. T. Freeman, *Removing camera shake from a single photograph*, ACM Trans. Graph. **25**, 787 (2006).

[46] J. Jia, Single image motion deblurring using transparency, in *CVPR*, 2007.

[47] L. Bar, B. Berkels, M. Rumpf, and G. Sapiro, A variational framework for simultaneous motion estimation and restoration of motion-blurred video, in *ICCV*, 2007.

[48] Q. Shan, W. Xiong, and J. Jia, Rotational Motion Deblurring of a Rigid Object from a Single Image, in *ICCV*, 2007.

[49] C. S. Won, K. Pyun, and R. M. Gray, Automatic object segmentation in images with low depth of field, in *ICIP (3)*, pages 805–808, 2002.

[50] L. Kovács and T. Szirányi, *Focus Area Extraction by Blind Deconvolution for Defining Regions of Interest*, PAMI **29**, 1080 (2007).

[51] J. Z. Wang, J. Li, R. M. Gray, and G. Wiederhold, *Unsupervised Multiresolution Segmentation for Images with Low Depth of Field*, PAMI **23**, 85 (2001).

[52] R. Datta, D. Joshi, J. Li, and J. Z. Wang, Studying aesthetics in photographic images using a computational approach, in *ECCV (3)*, pages 288–301, 2006.

[53] G. Burton and I. Moorhead, *Color and spatial structure in natural scenes*, Applied Optics **26**, 157 (1987).

[54] D. Field, *Relations between the statistics of natural images and the response properties of cortical cells*, Journal of the Optical Society of America A **4**, 2379 (1987).

[55] D. Tolhurst, Y. Tadmor, and T. Chao, *Amplitude spectra of natural images*, Ophthalmic Physiol Opt **12**, 229 (1992).

[56] P. Bex and W. Makous, *Spatial frequency, phase, and the contrast of natural images*, Journal of the Optical Society of America A **19**, 1096 (2002).

[57] D. Field and N. Brady, *Visual sensitivity, blur and the sources of variability in the amplitude spectra of natural scenes*, Vision Research **37**, 3367 (1997).

[58] B. Hansen and R. Hess, *Discrimination of amplitude spectrum slope in the fovea and parafovea and the local amplitude distributions of natural scene imagery*, Journal of Vision **6**, 696 (2006).

[59] S. Roth and M. J. Black, Fields of experts: A framework for learning image priors, in *CVPR (2)*, pages 860–867, 2005.

[60] E. W. Weisstein, Autocorrelation, MathWorld. http://mathworld.wolfram.com/Autocorrelation.html.

[61] C. Harris and M. Stephens, *A combined corner and edge detector*, Alvey Vision Conference **15** (1988).

[62] Y. Li, J. Sun, C.-K. Tang, and H.-Y. Shum, *Lazy snapping*, ACM Trans. Graph. **23**, 303 (2004).

[63] V. Kolmogorov and R. Zabih, *What Energy Functions Can Be Minimized via Graph Cuts?*, PAMI **26**, 147 (2004).

Appendix A

Blurred Edge Analysis

In the following, we give a detail analysis of the relationship between the blur direction and edge direction. First we modelled focal blur and directional motion blur using different Gaussian filters, respectively. Then analysis of blurring effects on edges are given to show the relationship of edge direction and blur direction. Through our analysis and modelling of blurring effect, we can derive the following attributes of the blurring effect:

A. The blurring effect of focal blur is isotropic, i.e. smoothing edges in every direction.

B. The blurring effect of motion blur is anisotropic. Edges with the same direction as the motion direction will be the least affected, while edges perpendicular to the motion direction will be blurred most severely.

C. Since the blur kernel shows no effect when applying to a flat region without illuminance changes. So flat regions contain no useful information for blur detection.

Let $u(x)$, as shown in Fig. A.1(a), be a step function along the y-axis. An edge is modelled as a step function $S(x, y) = Au^{\theta_e}(x) + B$ of unknown amplitude A and pedestal offset B, where $u^{\theta_e}(x)$ denotes a rotation of step function $u(x)$ with the origin as rotation center by an angle $(\theta_e - \frac{\pi}{2})$, as shown in Fig. A.1(b).

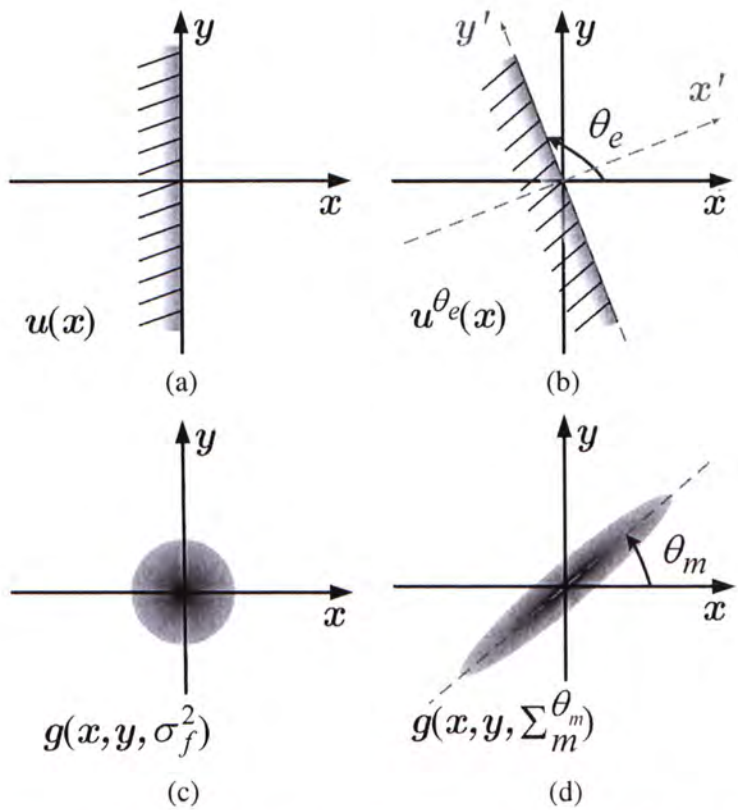


Figure A.1: Diagrams to illustrate the modelling of edges and the blur filters for focal and motion blur. (a) An illustration of 2D step function $u(x)$ along the y-axis. (b) The 2D illustration of $u^{\theta_e}(x)$, resulting from the rotation of $u(x)$ by an angle $(\theta_e - \frac{\pi}{2})$ with the origin as rotation center. An edge with angle θ_e between the positive x-axis is modelled based on $u^{\theta_e}(x)$. Also, the rotated coordinate system (x', y'), used in Equations (A.4) and (A.5), is shown in red. (c) An illustration of focal blur model, $g(x, y, \sigma_f^2)$, in shape of a circularly symmetric Gaussian filter. (d) An illustration of motion blur model, $g(x, y, \Sigma_m^{\theta_m})$, which is an ellipse-shaped Gaussian filter, with angle θ_m between its principle axis and the positive x-axis, denoting the overall motion direction.

For out-of-focus blur, if the aperture of the camera is near-circular, the image of any point source is a small disk, known as the Circle of Confusion(COC). An accurate focal blur model should not only describe the diameter of COC, but also the intensity distribution within the COC. However, since the diameter of COC varies dramatically even between neighboring pixels if they are projected from sperate 3D points with different depth. So here we modelled the focal blur as an isotropic (i.e. circularly symmetric) Gaussian blur filter, $g(x, y, \sigma_f^2) = \frac{1}{2\pi\sigma_f^2} e^{-(x^2+y^2)/2\sigma_f^2}$ of unknown scale constant σ_f, as shown in Fig. A.1(c). Sensor noise $n(x, y)$ is modelled as a stationary, additive, zero-mean white noise process with standard deviation σ_n. The complete focal blurred edge model is thus:

$$\begin{aligned} S_f(x,y) &= S(x,y) * g\left(x, y, \sigma_f^2\right) + n(x,y) \\ &= \frac{A}{2}\left[erf^{\theta_e}\left(\frac{x}{\sqrt{2}\sigma_f}\right) + 1\right] + B + n(x,y) \end{aligned} \tag{A.1}$$

where $*$ denote the convolution operator.

Motion blur, on the other hand, is much more difficult to model. Since too many undeterminable factors involved in estimating the motion trajectory, we modelled motion blur using a generalized Gaussian blur filter, $g(x, y, \Sigma_m^{\theta_m})$, with covariance matrix

$$\Sigma_m^{\theta_m} = \left(R(\theta_m)^T \begin{bmatrix} \sigma_{mx}^2 & 0 \\ 0 & \sigma_{my}^2 \end{bmatrix}^{-1} R(\theta_m) \right)^{-1} \tag{A.2}$$

where $R(\theta_m)$ is the rotation matrix which rotate the Gaussian filter by a counter-clockwise angle θ_m, and $\sigma_{mx}^2 \gg \sigma_{my}^2$. As illustrated in Fig. A.1(d) that, motion blur is modelled as an ellipse-shaped Gaussian with an angle θ_m between its principle axis and the positive x-axis. θ_m here is introduced to reflect the overall motion direction. And the blur shape may change gradually from ellipse, and long ellipse to slim column if we keep increasing the ratio of $\sigma_{mx}^2/\sigma_{my}^2$. The complete motion blurred edge model is thus:

$$S_m(x,y) = S(x,y) * g(x, y, \Sigma_m^{\theta_m}) + n(x,y) \tag{A.3}$$

If we first rotate the coordinate of $S_f(x,y)$ and $S_m(x,y)$ by angle $(\theta_e - \pi/2)$ to get $S_f(x',y')$ and $S_m(x',y')$, and take the first order derivative of $S_f(x',y')$ and $S_m(x',y')$ with respect to x' and y', according to the differentiation property of convolution, we have the gradient response

$$\begin{aligned} r_{x'}(x',y',\sigma_f) &= \frac{\partial}{\partial x'} S_f(x',y') \\ &= \left[\frac{\partial}{\partial x'} S(x',y')\right] * g(x',y',\sigma_f^2) \\ &= \left[\frac{\partial}{\partial x'} Au(x')\right] * g(x',y',\sigma_f^2) \\ &= \frac{A}{\sqrt{2\pi\sigma_f^2}} e^{-x'^2/2\sigma_f^2} \end{aligned} \tag{A.4}$$

$$\begin{aligned} r_{x'}(x',y',\Sigma_m^{\theta_m}) &= \frac{\partial}{\partial x'} S_m(x',y') \\ &= \left[\frac{\partial}{\partial x'} S(x',y')\right] \\ &\quad * g\left(x',y',\Sigma_m^{\theta_m-\theta_e+\pi/2}\right) \\ &= \frac{A}{\sqrt{2\pi\eta^2}} e^{-x'^2/2\eta^2} \end{aligned} \tag{A.5}$$

where $\eta^2 = \sin^2\theta\sigma_{my}^2 + \cos^2\theta\sigma_{mx}^2$, and $\theta = \theta_m - \theta_e + \pi/2$. Both $r_{x'}(x',y',\sigma_f)$ and $r_{x'}(x',y',\Sigma_m^{\theta_m})$ will attain their maxima on the y'-axis(Noise is not consider here). However, the blurring extent of gradient magnitude, which can be considered as the descending speed away from maxima points, is different between motion blur and focal blur. If we examine the distribution of these two gradient responses, they resemble Gaussian distributions up to a scale of a standard deviation. So the variance can be used to measure the blurring extent. We find that for focal blur, the variance of $r_{x'}(x',y',\sigma_f)$ only depend on the variance of focal blur σ_f, which means the heavier the focal blur, the heavier the edge would blur. However, for motion blur, the variance of $r_{x'}(x',y',\sigma_f)$ depend on a factor η, which is jointly decided by θ_m, θ_e, σ_{mx} and σ_{my}. We further analyze the distribution of η^2

$$\begin{aligned} \eta^2 &= \sin^2\theta\sigma_{my}^2 + \cos^2\theta\sigma_{mx}^2 \\ &= \sigma_{my}^2 + \left(\sigma_{mx}^2 - \sigma_{my}^2\right)\cos^2\theta \end{aligned}$$

It follows the distribution of $\cos^2\theta$. Since we have $\sigma_{mx}^2 \gg \sigma_{my}^2$, as defined in Equation (A.2), we can see that the value of η^2 is in the range of $[\sigma_{my}^2, \sigma_{mx}^2]$. It reaches its maxima σ_{mx}^2 when $\theta = n\pi$, and minima σ_{my}^2 when $\theta = n\pi + \pi/2$, where $n \in \mathcal{Z}$. This is easy to understand. When $\theta = n\pi$, i.e. $\theta_m - \theta_e = n\pi + \pi/2$, we have the motion blur filter which motion direction is perpendicular to the direction of the edge, thus it will cause the strongest blurring effect to the edge, with the maximum variance η^2. On the other hand, when $\theta = n\pi + \pi/2$, i.e. $\theta_m - \theta_e = n\pi$, we have the motion blur filter with the same motion direction as the direction of the edge. Thus the blurring effect to the edge would be the least noticeable, with the minimum variance η^2 to the gradient response.

From the analysis of blurred edges above, we can see that our conclusions on the blurring effects of different blur types and the attributes of blurring effect are correct.